

REMARKS

Prior to the present Amendment, claims 1-5, 7-22, 24-29 and 31-33 were pending in the application. Claims 10 and 13-15 are cancelled and new claims 34-39 have been added. Claims 1, 2, 3, 4, 11, 16, 18-22, 25, 29 and 32 have been amended herein. Thus, claims 1-5, 7-9, 11, 12, 16-22, 24-29, and 31-39 are presented for further consideration.

The specification has been amended to include new paragraphs 29-36 to include a description of the user interface of Fig. 8 and the operation thereof for initiating transferring a database from a server to a client in accordance with the present invention. Additionally, paragraph [0037] has been amended to include language stating that in one embodiment, the system of the present invention utilizes the standard Web (HTTP) protocol and the client is a standard Java Applet that runs on a standard Web browser enabled with Java and is loaded using a standard URL for improved firewall penetration.

No new matter has been added to the Application. Support for the amendments to the Specification and Claims is set forth in Fig. 8 of the Application and the U.S. Provisional Application Serial No. 60/441,604 from which the present Application claims priority and which is incorporated herein by reference.

Claims 1-5, 7-21, 25-29 and 31-33 are rejected under 35 U.S.C. § 112, second paragraph as allegedly being indefinite based on the term "and/or" recited in claims 1 and 25. Claims 1 and 25 have been amended herein to remove the term "and/or". Thus, the Examiner's rejection of the above-identified claims under 35 U.S.C. § 112, second paragraph should be withdrawn.

Claims 1-5, 7-22, 24-29 and 31-33 are rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over U.S. Patent No. 6,604,110 to Savage et al. (hereinafter "Savage") and U.S. Patent No. 7,065,541 to Gupta et al. (hereinafter "Gupta") and further in view of U.S. Patent Publication No. 2006/0277227 A1 of Britton et al. The rejection is traversed with respect to the claims as amended herein and reconsideration is respectfully requested.

Savage is directed to a method and apparatus for analyzing the relations, attributes, and rows or records of data stored within a source database to determine its metadata. The apparatus analyzes the source database in an elemental sequence that is defined by the logical and structural model of a data repository having

defined entity relationships, so as to provide a comprehensive metadata model of the source data. The detailed information provided by the metadata is sufficient to provide a generic description of the source data that can be used to generate program code for a plurality of enterprise data management (EDM) applications. (Savage, col. 4, ll. 56-67). Savage further discloses that the data repository architecture supports the manual entry of migration specifications including mapping the data migration from the source database into a new target database by defining the transformations of the source data required to migrate data from the original source tables into new target tables. (Savage, col. 22, ll. 55-65). Thus, the migration specification can be utilized to move the original source data into the target database by generating an ETL job based on the specification. (Savage, col. 23, ll. 8-10). Savage further recites that information may be needed from third parties to generate appropriate transfer protocol. (Savage, col. 23, ll. 16-27).

Thus, Savage discloses a system for the generation of executable software code for other third party systems to use for various data processing tasks which can include database migration. In the Savage system, metadata must be created, which may require user interaction throughout the process. Once the metadata has been created, it is used, in a separate step, to generate code, which is then used by a third party system for performing a task.

Accordingly, the Savage system requires substantial development effort in order to transfer a database from one location to another and does not provide a client-server system for the transfer of a source database to the client across various database types, vendors and operating systems without utilizing persons skilled in database art developing ETL tools to carry out the database transfer. Further, each database to be transferred must be handled separately, including creating an appropriate migration specification and generating an ETL job based on the migration specification.

Gupta discloses a method and system for migrating a database from a first server to a second server while continuing to provide transaction service with the database during the migration. An active database is copied to a target and updated at least one time. In one embodiment, updating occurs over decreasing time intervals. When the time intervals become sufficiently short, the transition to the target server is implemented by queuing transaction requests from the source server and then executing them on the target server. (Gupta, Abstract).

The Gupta system requires the source and target database be of the same type and provided by the same vendor including a previous database to be overwritten and/or the version and release of the RDBMS at the target is compatible with the source. (See Gupta, col. 6, ll. 10-20). Gupta clearly states the "[s]ervers 4 and 6 generally are provided as application servers that include the particular ASP's application capable of accessing and modifying database 8." (See Gupta, col. 3, ll. 64-66). Further Gupta requires preparing the target which includes "assuring storage requirements are met at a target data storage device, or assuring a database environment is present (e.g. a previous database to be overwritten is present) and/or that the version/release level of the RDBMS is compatible." (See Gupta, col. 6, ll. 10-20).

Britton et al. teaches a method for enterprise application integration that uses software "connectors" that can be downloaded to provide interfaces to respective disparate database systems. The connectors can, for example, translate between a native language (or API) of the respective database systems and an internal language/protocol of the enterprise application integration system. The connectors can utilize a scripting language to access the respective database systems. In another aspect, Britton et al. provides for storing data accessed from the database systems in a central data store, for example as resource definition framework (RDF) triplets. Further, the connectors can query the respective database systems based on requests from the data store, a framework server, or a user.

The present invention transfers databases securely over the Web and across database types, vendors and operating systems with a maximum of ease and speed and does not require firewall modification or manual software development or installation by a user. The user is not required to develop or load any software nor provide any programming or scripting to carry out the transfer. The system of the present invention provides a graphical user interface for a user to operate the system and requires the user to identify only the source database to be transferred and a target database to receive the transferred data. Thereafter, the user need only to initiate the transfer via a start button. Nothing further is required from the user. As discussed in detail below with respect to the pending claims, the teachings of Gupta, Savage and Briton et al. even if combined, do not teach or suggest the claimed system and method.

Applicants' claim 1 as amended herein, recites a client-server system for transferring a database from a server to a client over the Web, the server including a Web server having a source database of a first database type accessible thereto, the client automatically downloadable from the server and having a Web browser connectable to the Web for communicating with the Web server. The client connectable to a database system for storing a copy of the source database in a target database of a second database type. The client server system including a graphical user interface operable at the client for causing the generation and population of the target database at the client. Claim 1 further recites, initiation of a database transfer via the graphical user interface requiring only identifying the source database, identifying the target database, and use of a selector to initiate the transfer. Further, claim 1 recites the server having a programming interface including an interface corresponding to each of a plurality of known databases stored thereon. The server being programmed to automatically identify and load the programming interface corresponding to the source database for accessing the source database and retrieving the metadata and at least a portion of the data and storing the retrieved data in at least one data object. Amended claim 1 further recites the server sending the metadata and the at least one data object to the client over the Web via a Web protocol over standard Web channels such that firewall modification is not required at the client. Amended claim 1 further recites the client including a processor for automatically downloading and installing an interface corresponding to the target database, receiving the metadata and the at least one data object from the server, generating the target database using the metadata, and populating the target database with the data from the at least one data object including mapping the data types from the source database of the first type to that of the target database of the second type.

The client-server system of claim 1 providing for the transfer of a source database of a first database type to the client and target database of a second database type across various database vendors and operating systems without user interaction following initiation of the transfer via the graphical user interface. The system being automated such that no programming or scripting is required for initiating or carrying out the transfer.

The cited combination of Savage, Gupta and Britton et al. references do not teach or suggest a client-server system for transferring a database from a source

database of a first database type to a target database of a second database type over the Web without requiring programming or scripting for initiating or carrying out the transfer as recited in Applicants' amended claim 1 for at least the following reasons:

First, nothing in the combination of the Gupta, Savage and Britton et al. references teaches or suggests transfer of a source database from a server to a client over the Web via a Web protocol over standard Web channels such that firewall modification is not required at the client as recited in amended claim 1. Gupta nor Savage contemplate transfer of a source database to a client over the Web via a Web protocol. Gupta suggests database transmission over a network including the Internet, however, nothing in Gupta provides for transferring a database via a Web protocol over standard Web channels such that firewall modification is not required at the client. Britton et al. provides results of a database query using a Web browser, yet does not teach or suggest transferring a database from a server to a client. Therefore, transferring a database from a server to a client over the Web via a Web protocol such that firewall modification is not required at the client as recited in amended claim 1 is not taught or suggested by either of Savage, Gupta or Britton et al. or the combination thereof.

Secondly, claim 1 has been amended herein to recite transferring a source database of a first database type to a target database of a second database type. Clearly, nothing in Gupta, Savage and Briton et al. provide for transferring a database from one location to the other across various database types, vendors and operating systems without development effort as stated by the Examiner. (See Office Action, p. 4, ll. 4-6). Although Gupta recites "... each data storage device of Figs. 1A-B) may comprise any known type of data storage system and/or transmission media", nothing in Gupta teaches or suggests that the migration system of Gupta provides for the transfer of data from a source database of a first database type to a target database of a second database type across various database vendors and operating systems as recited in Applicant's amended claim 1. (Gupta, col. 4, l. 66, col. 5, l. 1). Gupta clearly requires that [s]ervers 4 and 6 (source and target servers, respectively) are provided as application servers that include the particular ASP's application capable of accessing and modifying database 8. (Gupta, col. 3, ll. 64-66). Accordingly, the Gupta migration system utilizes the ASP for the source database at both the server and the client. Further, Gupta recites preparation of the

target data storage device including requiring either a database to be overwritten and/or the version release level of the RDBMS is compatible with the source. (See Gupta, col. 6, ll. 10-20). Regarding preparing a target database, Gupta further recites:

[i]n one embodiment, this step may entail making a backup of active database 8 from the source and communicating it to the target. A database does not need to exist on the target in advance. When a database backup is restored to an existing database, the existing database inherits the alias and database names of the existing database. When restoring to a nonexistent database, the new database will be created with an alias and database name specified by a target-database-alias parameter.

Gupta teaches backup and restore processes enabled using tools external to the Gupta device. Typically, making and restoring backups is done with proprietary tools specific to each database. Since Gupta does not disclose other means for making and restoring backups, the Gupta invention is dependent on external tools for making and restoring backups and therefore is limited to making and restoring backups associated with like database systems. Additionally, Figures 1A and 1B of Gupta both show a data connection directly from one database to another, which would be appropriate only for a back-up or restore process between like database systems. Direct database connections, as shown in Figs. 1A and 1B of Gupta, can not be made across database systems of different types. Thus, consistent with the specification and figures of Gupta, the Gupta migration system requires the source and target database to utilize the same ASP and therefore does not provide for the transfer of data from a source database of a first database type to a target database of a second database type across various database vendors and operating systems as recited in amended claim 1.

Moreover, nothing in the combination of Gupta, Savage and Britton et al. teach or suggest mapping data types from the source database of a first database type to that of the target database of a second database type as recited in amended claim 1. Since the cited references do not provide for transferring a database across vendors, there is no disclosure nor contemplation as to mapping data types from the source database to that of the target database.

The Examiner has acknowledged that Savage does not teach or suggest the transfer of data from a source to a target database across various database types, vendors and operating systems as recited in Applicant's claim 1. (See Office Action,

p. 3). Accordingly, for at least the reasons set for above, even if combined, the teachings of Savage, Gupta and Britton et al. do not teach or suggest the transfer of a database from a source to a client across various database types, vendors, and operating systems.

Further, amended claim 1 recites a graphical user interface (GUI) operable at the client for causing the generation and population the target database, initiation of a database transfer via the GUI requiring only identifying the source and target databases and use of a selector to initiate the database transfer. The client-server system of claim 1 provides for the transfer of a source database to the client without user interaction following initiation of the transfer request via the GUI, wherein the system is automated such that no programming or scripting is required for initiating or carrying out the transfer. The user is not required to have any knowledge of the source database other than the name thereof. Neither of the Gupta, Savage, nor Britton et al. references or the combination thereof, teach or suggest a GUI operable at the client for causing the generation and population of a target database, wherein initiation of a database transfer is provided via the GUI requiring only identifying the source and target databases and use of a selector to initiate the database transfer as recited in amended claim 1.

Moreover, amended claim 1 recites wherein the client automatically downloads and installs an interface corresponding to the target database. As set forth above, Gupta requires that [s]ervers 4 and 6 (source and target servers, respectively) are provided as application servers that include the particular ASP's application capable of accessing and modifying database 8. (Gupta, col. 3, ll. 64-66). Clearly, the Gupta migration system requires the ASP for the source database at both the server and the client, thus a common database vendor for the source and target databases is required. Savage requires customized tools to access a target database. Britton et al. does not teach or suggest a target database nor copying the source database at the client and does not disclose automatically downloading and installing an interface corresponding to a target database for copying the source database at the client as recited in Applicant's claim 1. Thus, the combination of Gupta, Briton and Savage does not teach or suggest the client automatically downloading and installing a interface corresponding to a target database as set forth in Applicants' claim 1.

To establish a *prima facie* case of obviousness for a claimed invention, all of the claim limitations must be taught or suggested by the prior art. *In re Royka*, 490 F.2d 981, 180 USPQ 580 (CCPA 1974). Here, as set forth above, nothing in the combination of the Savage, Gupta and the Britton et al. references teaches or suggests: 1) a client-server system for transferring a database from a server to a client over the Web via a Web protocol such that firewall modification at the client is not required; 2) transfer of a source database to a client across various database types, vendors and operating systems without user interaction following initiation of the transfer; 3) providing a GUI operable at the client for causing the generation and population of the target database wherein initiation of a database transfer requires only identifying the source and target databases and use of a selector to initiate the transfer; 4) a client automatically downloadable from the server and installable in a Web browser connectable to the Web for communication with the Web server; and 5) the client automatically downloading and installing an interface corresponding to the target database for use in generating and populating the target database. Because the combination of Savage, Gupta and Britton et al. does not teach or suggest the above-identified features, it cannot be maintained that the cited combination teach or suggest each and every limitation of amended claim 1. Accordingly, for at least the above-identified reasons, amended claim 1 is not obvious under 35 U.S.C. § 103(a) over Savage in view of Gupta and further in view of Britton et al.

Claims 2-5, 7-9, 11, 12, 16-21 and 34-38 depend either directly or indirectly from claim 1 and thereby include all of the limitations of claim 1 and also include additional limitations. Since, claim 1 is not obvious under 35 U.S.C. § 103(a) over the combination of Savage, Gupta and Britton et al. for at least the above-identified reasons, the above-identified dependent claims are also not obvious over the combination of Savage, Gupta and Britton et al.

Claim 22 is rejected under 35 U.S.C. § 103(a) as allegedly being unpatentable over Savage in view of Gupta and Britton et al.

Claim 22 as amended herein recites a data access application for use in transferring a source database from a server to a client over the Web, the source database of a first database type accessible to the server having metadata and database data stored therein. The server associating a structure object with the metadata and a data object with the database data. A client automatically

downloadable from the server and installable in a Web browser connectable to the Web for communication with the Web server, the client connectable to a database system for storing a copy of the source database in a target database of a second database type. The client being configured for receiving a programmatically generated HTTP transfer request for transferring the source database from the server to the client. The client automatically initiating the transfer upon receipt of the transfer request. Claim 22 also recites wherein the client-server system provides for the transfer of the source database of the first database type to the client and target database of the second database type across various database vendors and operating systems.

Nothing in the combination of Savage, Gupta and Britton et al. disclose a data access application for use in transferring a database from a server to a client over the Web including a client automatically downloadable and installed in a Web browser as recited in claim 22. Further, nothing in the combination of Savage, Gupta and Britton et al. teach or suggest the client configured for receiving a programmatically generated HTTP transfer request for transferring the source database from the server to the client wherein the client automatically initiates the transfer upon receipt of a transfer request as recited in amended claim 22.

Further, as set forth above with respect to claim 1, nothing in the combination of Savage, Gupta and Britton et al. references teaches or suggests transferring a structure and data object corresponding to a source database to a client over the Web via a Web protocol over standard Web channels such that firewall modification is not required at the client as recited in amended claim 22.

Accordingly, for at least the above-identified reasons, amended claim 22 is not obvious under 35 U.S.C. § 103(a) over Savage in view of Gupta and Britton et al.

Claim 24 depends from claim 22 as is therefore also deemed patentable over the combination of Savage, Gupta and Britton et al. for at least the above-identified reasons.

Amended claim 25 recites a method for transferring a database from a server to a client over the Web, the server connected to the Web and including a Web server. Similar to claim 1, the method of claim 25 includes claim language directed to method steps corresponding to each of the limitations of amended claim 1 identified above which are not taught or suggested by the cited combination of Savage, Gupta, and Britton et al. references. Accordingly, for at least the reasons set

forth above with respect to claim 1, claim 25 is not obvious under 35 U.S.C. § 103(a) over Savage in view of Gupta and Britton et al. Therefore claim 25 should be allowed and this action is respectfully requested.

Claims 26-29 and 31-33 depend from claim 25 and therefore are also deemed not obvious over the combination of Savage, Gupta and Britton et al. for at least the reasons set forth above with respect to claim 25.

Conclusion

In view of the foregoing, it is respectfully submitted that pending claims 1-5, 7-9, 11, 12, 16-22, 24-29, and 31-39 are now in condition for allowance. All issues raised by the Examiner having been addressed, an early action to that effect is earnestly solicited.

Petition for Extension of Time to Respond

Pursuant to 37 CFR 1.136(a), Applicant hereby requests a three-month extension for filing a reply to the Final Office Action of March 8, 2007, thereby extending the period to respond through September 8, 2007. Authorization is hereby given to charge \$510 to our Deposit Account No. 13-0235 to cover the fee for the three month extension under 37 CFR 1.17(a)(1).

No additional fees or deficiencies in fees are believed to be owed. However, authorization is hereby given to charge our Deposit Account No. 13-0235 in the event any such fees are owed.

Respectfully submitted,

By /Donald J. MacDonald/
Donald J. MacDonald
Registration No. 42,823
Attorney for Applicant

McCormick, Paulding & Huber LLP
CityPlace II, 185 Asylum Street
Hartford, CT 06103-3402
Tel. 860-549-5290
Fax 413-733-4543